2018 Annual American Control Conference (ACC)
June 27–29, 2018. Wisconsin Center, Milwaukee, USA

Robust Temporal Logic Inference for Hybrid System Observation– An Application on Occupancy Detection of Smart Buildings

Zhe Xu, Yi Deng and Agung Julius

Abstract— **In modern smart buildings modeled as hybrid systems, occupancy detection can be cast as observing the discrete states of a hybrid system using the available discrete and continuous system outputs. In this paper, we present a method to construct observers of the hybrid system to distinguish between different locations of the hybrid system by inferring metric temporal logic (MTL) formulae from the simulated trajectories. We first approximate the system behavior by simulating finitely many trajectories with time-robust tube segments around them. These time-robust tube segments account for both spatial and temporal uncertainties that exist in the hybrid system with initial state variations. The inferred MTL formulae classify different time-robust tube segments and thus can be used for classifying the hybrid system behaviors in a provably correct fashion. We implement our approach on a model of a smart building testbed to distinguish two cases of room occupancy.**

I. Introduction

In modern smart buildings, various continuous states such as the temperature, humidity and discrete states such as the air conditioning states have made the system a hybrid system. In a hybrid system, the continuous state keeps flowing in a location (also called a mode or discrete state) until an event is triggered. Then it jumps to a target location and flows continuously again according to possibly different dynamics. As there are increasingly growing interest in finding ways to accurately determine localized building or room occupancy in real time, traditional methods seldom apply to multiple dynamics in a hybrid system.

Presently, there are mainly two categories of approaches for occupancy detection or estimation (e.g. detecting or estimating the number of people in a room). The first category relies on the learning-based techniques such as decision trees [1] or support vector regression [2] to find features of different occupancy states from data gathered from various sensors. The second category relies on the mathematical model of systems as they compare available measurements with information analytically derived from the system model [3]. For hybrid systems, the main challenge of the model-based occupancy detection is due to the difficulty in capturing the combined continuous and discrete measurements.

In this paper, we propose an approach that utilizes both the learning-based techniques and the model-based methods for hybrid system occupancy detection. We mainly focus on distinguishing between different occupancy states and observing the location of the modeled hybrid system at any time. For the learning aspect, there has been a growing interest in learning (inferring) dense-time temporal logic formulae from system trajectories [4], [5], [6], [7], [8], [9], [10]. We infer dense-time temporal logic formulae from the temperature and humidity sensor data as dense-time temporal logics can effectively capture the time-related features in the transient period when people enter a room. In the meantime, we also utilize the model information so that the MTL formula that classifies the finite trajectories we simulated (or gathered) also classifies the infinite trajectories that differ from the simulated trajectories by a small margin in both space and time. In our previous work in [11], we have performed classification for trajectories generated from switched systems, which have spatial uncertainties due to initial state variations. In this paper, we extend the results to hybrid systems and we classify time-robust tube segments around the trajectories so that the inferred MTL formula can classify different system behaviors when both the spatial and the temporal uncertainties exist due to initial state variations in a hybrid system.

In inferring a temporal logic formula that classifies different system behaviors, we can further design an observer for determining the location of the hybrid system at any time. Our previous work [12] results in a hybrid observer that estimates both the discrete and continuous states constantly. The observer only uses the discrete outputs generated by the hybrid system's observable events and their timing information as its input, and thus is referred to as the basic observer in this paper. Based on [12], we utilize the inferred MTL formula from the MTL classifier to refine the basic observer and the obtained observer is referred to as the refined observer.

II. Preliminaries

A. Hybrid Automaton

A hybrid autonomous system is defined to be a 5-tuple $\mathcal{H} = (\mathcal{L}\times\mathcal{X}, \mathcal{L}^0\times\mathcal{X}^0, \mathcal{F}, \mathcal{E}, Inv)$ [13]:

- $\mathcal{L}\times\mathcal{X}$ is a set of hybrid states (ℓ, x), where $\ell\in\mathcal{L}$ is discrete state (location), and $x\in\mathcal{X}$ is continuous state.
- $\mathcal{L}^0\times\mathcal{X}^0\subset\mathcal{L}\times\mathcal{X}$ a set of initial states.
- $\mathcal{F} = \{f_\ell | \ell\in\mathcal{L}\}$ associates with each location $\ell\in\mathcal{L}$ the autonomous continuous time-invariant dynamics, $f_\ell : \dot{x} = f_\ell(x)$, which is assumed to admit a unique global solution $\xi_\ell(\tau, x^0_\ell)$, where ξ_ℓ satisfies $\frac{\partial\xi_\ell(\tau,x^0_\ell)}{\partial\tau} =$

Zhe Xu and Agung Julius are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, USA, E-mail: xuz8@rpi.edu, juliua2@rpi.edu. Yi Deng is with East China Institute of Computing Technology, Shanghai, China, E-mail: dengyi267@gmail.com. This research was partially supported by the National Science Foundation through grants CNS-1218109, CNS-1550029 and CNS-1618369.

$f_\ell(\xi_\ell(\tau, x_\ell^0))$, and $\xi_\ell(0, x_\ell^0) = x_\ell^0$ is the initial condition in ℓ.

- $Inv : \mathcal{L} \to 2^{\mathcal{X}}$ associates an invariant set $Inv(\ell) \subset \mathcal{X}$ with each location. Only if the continuous state satisfies $x \in Inv(\ell)$, can the discrete state be at the location ℓ.
- $\mathcal{E}$ is a set of events. In each location ℓ, the system state evolves continuously according to f_ℓ until an event $e = (\ell, \ell', g, r), e \in \mathcal{E}$ occurs. The event is guarded by $g \in Inv(\ell)$. Namely, a necessary condition for the occurrence of e is $x \in g$. After the event, the state is reset from (ℓ, x) to $(\ell', r(x))$, where $r(x)$ is the reset initial state of x.

When a hybrid system runs, the system state alternately flows continuously and triggers events in $\mathcal{E}$. For convenience, we also define an initialization event $e^0 \notin \mathcal{E}$. Then a trajectory of the system can be defined as a sequence:

Definition 1 (Trajectory): A trajectory of a hybrid system H is denoted as

$$\rho = \{(e^m, \ell^m, x_{\ell^m}^0, \tau^m)\}_{m=0}^N,$$

where

- $\forall m \geq 0$, $(\ell^m, x_{\ell^m}^0) \in \mathcal{L} \times \mathcal{X}$ are the (reset) initial states;
- $\forall m \geq 0$, $\tau^m \in \mathbb{R}_{\geq 0}$ (nonnegative real), and $\forall \tau \in [0, \tau^m]$, $\xi_{\ell^m}(\tau, x_{\ell^m}^0) \in Inv(\ell^m)$;
- $\forall m \geq 1$, $e^m = (\ell^{m-1}, \ell^m, g^m, r^m)$, $\xi_{\ell^{m-1}}(\tau^{m-1}, x_{\ell^{m-1}}^0) \in g^m$, $x_{\ell^m}^0 = r^m(\xi_{\ell^{m-1}}(\tau^{m-1}, x_{\ell^{m-1}}^0))$, i.e. $(\ell^m, x_{\ell^m}^0)$ is the reset initial state for $(\ell^{m-1}, \xi_{\ell^{m-1}}(\tau^{m-1}, x_{\ell^{m-1}}^0))$.

Each event $e \in E$ has an output symbol $\psi(e)$ that can be observable or unobservable. An unobservable output symbol $\psi(e)$ is specifically denoted as ϵ.

B. Robust Neighborhood Approach

In this section, we briefly review the robust neighborhood approach [14], which is based on the approximate bisimulation theory [15]. The robust neighborhood approach [14] is to compute a neighborhood around a simulated initial state, such that any trajectory initiated from the neighborhood will trigger the same event sequence as the simulated trajectory, and the continuous state always stays inside a neighborhood around the continuous state of the simulated one.

Definition 2: $\Phi_\ell : Inv(\ell) \times Inv(\ell) \to \mathbb{R}$ is an autobisimulation function for the dynamics of hybrid system $\mathcal{H}$ at location ℓ, if it satisfies

$$\Phi_\ell(x, \tilde{x}) > 0, \forall x, \tilde{x} \in Inv(\ell), x \neq \tilde{x},$$
$$\Phi_\ell(x, x) = 0, \forall x \in Inv(\ell),$$
$$\frac{\partial \Phi_\ell(x, \tilde{x})}{\partial x} f_\ell(x) + \frac{\partial \Phi_\ell(x, \tilde{x})}{\partial \tilde{x}} f_\ell(\tilde{x}) \leq 0.$$

From Definition 2, Φ_ℓ can be used to bound the divergence of continuous state trajectories. If we define the level set

$$B_\ell(\gamma_\ell, \xi_\ell(\tau, x_\ell^0)) \triangleq \{x | \Phi_\ell(x, \xi_\ell(\tau, x_\ell^0)) < \gamma_\ell\}. \quad (1)$$

then we can easily conclude that the value of Φ_ℓ is nondecreasing along any two trajectories of the system at location ℓ, i.e. for any initial state $\tilde{x}_\ell^0 \in B_\ell(\gamma_\ell, x_\ell^0)$ and $\tau > 0$, $\xi_\ell(\tau, \tilde{x}_\ell^0) \in B_\ell(\gamma_\ell, \xi_\ell(\tau, x_\ell^0))$.

Let $e = (\ell, \ell', g, r)$ be an event triggered by a trajectory initiated from x_ℓ^0. If we want all the trajectories initiated from within $B_\ell(\gamma_\ell, x_\ell^0)$ to avoid triggering a different event $e' = (\ell, \ell'', g', r')$, then we can let

$$\gamma_\ell \leq \inf_{y \in g'} \inf_{\tau \in [0, \bar{\tau}]} \Phi_\ell(\xi_\ell(\tau, x_\ell^0), y), \quad (2)$$

where $\bar{\tau}$ is an upper bound of the time for trajectories initiated from $B_\ell(\gamma_\ell, x_\ell^0)$ to transition out of ℓ (for details on methods for estimating $\bar{\tau}$, see [14]). Then for any $\tilde{x}_\ell^0 \in B(\gamma_\ell, x_\ell^0), \tau \in [0, \tau]$, we have that $\xi_\ell(\tau, \tilde{x}_\ell^0)$ cannot reach g' and thus trigger e'.

Let $\rho = \{(e^m, \ell^m, x_{\ell^m}^0, \tau^m)\}_{m=0}^N$ denote the simulated trajectory. We can compute robust neighborhoods $B_{\ell^m}(\gamma_{\ell^m}, x_{\ell^m}^0)$ around the (reset) initial continuous states $x_{\ell^m}^0$ of ρ such that the property below holds.

Proposition 1: For any initial state $(\ell^0, \tilde{x}_{\ell^0}^0) \in \{\ell^0\} \times B_{\ell^0}(\gamma_{\ell^0}, x_{\ell^0}^0)$ and any trajectory $\tilde{\rho} = \{(e^m, \ell^m, \tilde{x}_{\ell^m}^0, \tilde{\tau}^m)\}_{m=0}^N$ that triggers the same event sequence with the simulated trajectory ρ, there exist $\tau_{lead}^m, \tau_{lag}^m > 0$ $(0 \leq m \leq N-1)$ such that

- for all $0 \leq m \leq N-1$, $\tilde{x}_{\ell^m}^0 \in B_{\ell^m}(\gamma_{\ell^m}, x_{\ell^m}^0)$, $\tilde{\tau}^m \in [\tau^m - \tau_{lead}^m, \tau^m + \tau_{lag}^m]$, and $\Phi_{\ell^m}(\xi_{\ell^m}(t, x_{\ell^m}^0), \xi_{\ell^m}(t, \tilde{x}_{\ell^m}^0)) \leq \gamma_{\ell^m}$ for all $t \in [0, \tilde{\tau}^m]$;
- $\tilde{x}_{\ell^N}^0 \in B_{\ell^N}(\gamma_{\ell^N}, x_{\ell^N}^0)$, and $\Phi_{\ell^N}(\xi_{\ell^N}(t, x_{\ell^N}^0), \xi_{\ell^N}(t, \tilde{x}_{\ell^N}^0)) \leq \gamma_{\ell^N}$ for all $t \in [0, \min(\tilde{\tau}^N, \tau^N)]$.

We simulate trajectories from the initial set $\mathcal{L}^0 \times \mathcal{X}^0$ and perform robust neighborhood computation. We denote $\rho_k = \{(e_k^m, \ell_k^m, x_{\ell_k^m}^0, \tau_k^m)\}_{m=0}^{N_k}$ as the kth simulated trajectory ($k = 1, 2, \dots$). The robust neighborhood around the (reset) initial state for the segment m of ρ_k is the following:

$$B_{\ell_k^m}(\gamma_{\ell_k^m}, x_{\ell_k^m}^0) = \{x | \Phi_{\ell_k^m}(x_{\ell_k^m}^0, x) < \gamma_{\ell_k^m}\}, \quad (3)$$

where $\Phi_{\ell_k^m}$ is the bisimulation function in location ℓ_k^m, and $\gamma_{\ell_k^m}$ is the radius of the computed robust neighborhood. The initial set can be covered by the robust neighborhoods around the initial states of the finitely simulated trajectories if:

$$\mathcal{L}^0 \times \mathcal{X}^0 \subset \bigcup_k \{\ell_k^0\} \times B_{\ell_k^0}(\gamma_{\ell_k^0}, x_{\ell_k^0}^0). \quad (4)$$

III. Robust Temporal Logic Inference for Classification with Spatial and Temporal Uncertainties

In this section, we present the robust temporal logic inference framework for classification that accounts for both spatial and temporal uncertainties. According to our previous work in [12], given a trajectory simulated from an initial state, the possible discrete state (current location) can be estimated at any time by an observer for any trajectory initiated from a neighborhood around the simulated initial state. There are two notions of time in [12], one is the **external time** that can be read from an external timer and is reset to zero every time the constructed observer updates its states, the other one is the **clock time** that is associated with each trajectory which is reset to zero every time the trajectory enters a new location. **In this paper, we use t to denote the external time and τ to denote the clock time.** As different trajectories may reach the guards or leave an invariant set at different times, the clock time that is associated with each trajectory has temporal uncertainties. As the clock time is reset to zero when the trajectory enters a new location, the clock time is also associated with each location the trajectory enters. It can be seen that τ in $\xi_{\ell_k^m}(\tau, x^0_{\ell_k^m})$ is the clock time associated with location ℓ_k^m (the location corresponding to the mth segment of the kth trajectory ρ_k). We denote s as the set of possible observer states at the current time. At the external time t, we denote $(k,m)[\bar{a},\bar{b}] \in s$ if ℓ_k^m is possible as the current location and the clock time τ in location ℓ_k^m has temporal uncertainty $\tau \in [t+\bar{a}, t+\bar{b}]$ (see Proposition 2 in [12]). For example, if the observer state $s^1 = \{(1,1)[10,12], (2,1)[-6,2]\}$, $s^2 = \{(2,1)[2,10], (2,2)[-8,0]\}$, and the observer state update is $s^1 \xrightarrow{\epsilon[8]} s^2$ (here $\epsilon[8]$ means no event is observed for 8 time units), then at external time $t \in [0,8)$, the state could be in location ℓ_1^1 or ℓ_2^1, the clock time τ_1^1 for location ℓ_1^1 has temporal uncertainty $\tau_1^1 \in [t+10, t+12]$, the clock time τ_2^1 for location ℓ_2^1 has temporal uncertainty $\tau_2^1 \in [t-6, t+2]$. The external time t is reset to 0 when s^1 is updated by s^2 and at the new external time t, the state could be in location ℓ_2^1 or ℓ_2^2, the clock time τ_2^1 for location ℓ_2^1 has temporal uncertainty $\tau_2^1 \in [t+2, t+10]$, the clock time τ_2^2 for location ℓ_2^2 has temporal uncertainty $\tau_2^2 \in [t-8, t]$. **In sum, after an observation is made, the external time t has no temporal uncertainties, while the clock time τ has temporal uncertainties**.

Note that when $\bar{a}$ in $(k,m)[\bar{a},\bar{b}]$ is negative, it actually represents "latent" states that are currently in other locations. For example, $(2,2)[-8,0]$ means at the external time t ($t < 8$), the hybrid system state may have already been at location ℓ_2^2 for τ time units (τ is the positive clock time in location ℓ_2^1, $\tau \in [0,t]$), but may also be at location ℓ_2^1 and will enter location ℓ_2^2 at the next $(-\tau)$ time unit (τ is the "virtual" negative clock time in location ℓ_2^2, $\tau \in [t-8, 0]$). To account for the negative times, we allow τ in the notation $\xi_{\ell^m}(\tau, x^0_{\ell^m})$ to be negative (the valuation of $\xi_{\ell^m}(\tau, x^0_{\ell^m})$ when $\tau < 0$ can be any vector in $\mathbb{R}^n$) to represent the "virtual" negative clock time when the state is not in location ℓ^m at the current time but will enter location ℓ^m at a future time $(-\tau)$.

The basic syntax and semantics of the Metric Temporal Logic can be found in [16]. We use $\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi, c)$ to denote the extended robustness degree of a trajectory segment $\xi_{\ell^m}(\tau, x^0_{\ell^m})$ with respect to a classification label c ($c \in \{1,-1\}$) and an MTL formula ϕ evaluated at a certain external time corresponding to the clock time τ for the ith trajectory (τ can be positive or negative). $\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi, c)$ is defined recursively via the following extended quantitative semantics:

$$\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \mu, c) = \begin{cases} \mathbf{Dist}_d(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \mathcal{O}(\mu)), & \text{if } \tau \geq 0, \\ \infty, & \text{if } \tau < 0 \text{ and } c = 1, \\ -\infty, & \text{if } \tau < 0 \text{ and } c = -1, \end{cases}$$

$$\begin{aligned} &\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \neg\phi, c) = -\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi, -c), \\ &\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi_1 \wedge \phi_2, c) = \min(\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi_1, c), \\ &\qquad \hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi_2, c)), \\ &\hat{r}(\xi_{\ell^m}(\tau, x^0_{\ell^m}), \phi_1 \mathcal{U}_I \phi_2, c) = \max_{\tau' \in (\tau + I)} \min\{\hat{r}(\xi_{\ell^m}(\tau', x^0_{\ell^m}), \\ &\qquad \phi_2, c), \min_{\tau'' \in [\tau, \tau')} \hat{r}(\xi_{\ell^m}(\tau'', x^0_{\ell^m}), \phi_1, c)\}. \end{aligned}$$

where

$$\mathbf{Dist}_d(x, \mathcal{O}(\mu)) \triangleq \begin{cases} -\inf\{d(x,y) | y \in cl(\mathcal{O}(\mu))\}, \text{if } x \notin \mathcal{O}(\mu), \\ \inf\{d(x,y) | y \in \mathcal{X} \setminus \mathcal{O}(\mu)\}, \text{ if } x \in \mathcal{O}(\mu), \end{cases}$$

is the signed distance from x to the set $\mathcal{O}(\mu)$ (the set of states that satisfy the atomic proposition μ), d is a metric on $\mathcal{X}$ and $cl(\mathcal{O}(\mu))$ denotes the closure of the set $\mathcal{O}(\mu)$. It can be seen that for atomic predicate μ, the extended robustness degree of the trajectory $\xi_{\ell^m}(\tau, x^0_{\ell^m})$ with respect to μ evaluated at a negative clock time (when $\tau < 0$) always has the same sign as the classification label, which is consistent with our purpose of classification as the classification result should not be affected by a state that does not appear yet at the evaluation time.

Definition 3: The time-robust tube segment at external time t corresponding to location ℓ_k^m and an interval $[t+\bar{a}, t+\bar{b}]$, denoted as $R_{tube}(k, m, [t+\bar{a}, t+\bar{b}])$, is defined as follows:

$$R_{tube}(k,m,[t+\bar{a},t+\bar{b}]) = \{(\tau, \xi_{\ell_k^m}(\tau, \tilde{x}^0_{\ell_k^m})) \mid \tau \in [t+\bar{a}, t+\bar{b}],\ \xi_{\ell_k^m}(\tau, \tilde{x}^0_{\ell_k^m}) \in B_{\ell_k^m}(\gamma_{\ell_k^m}, \xi_{\ell_k^m}(\tau, x^0_{\ell_k^m})) \text{ if } \tau \geq 0\},$$

As $B_{\ell_k^m}(\gamma_{\ell_k^m}, \xi_{\ell_k^m}(\tau, x^0_{\ell_k^m}))$ is obtained through the robust neighborhood approach, the time-robust tube segment can be also expressed as

$$\begin{aligned} &R_{tube}(k,m,[t+\bar{a},t+\bar{b}]) = \{(\tau, \xi_{\ell_k^m}(\tau, \tilde{x}^0_{\ell_k^m})) \mid \\ &\tau \in [t+\bar{a}, t+\bar{b}], \tilde{x}^0_{\ell_k^m} \in B_{\ell_k^m}(\gamma_{\ell_k^m}, x^0_{\ell_k^m})\}, \end{aligned}$$

Definition 4: Given a labeled set of trajectory segments $\{(\xi_{\ell_{k_i}^{m_i}}(\tau_i(t), x^0_{\ell_{k_i}^{m_i}}), c_i)\}_{i=1}^N$ (the external time t flows from 0 to T, $\tau_i(t)$ is the corresponding clock time for the ith trajectory, $\tau_i(t)$ can be positive or negative) from a hybrid system $\mathcal{H}$, $c_i \in \{1,-1\}$ is the classification label, an MTL formula ϕ evaluated at external time 0 perfectly classifies the trajectory segments with label $c_i = 1$ and the trajectory segments with label $c_i = -1$ if the following condition is satisfied:
$\hat{r}(\xi_{\ell_{k_i}^{m_i}}(\tau_i(0), x^0_{\ell_{k_i}^{m_i}}), \phi, c_i) > 0$, if $c_i = 1$; $\hat{r}(\xi_{\ell_{k_i}^{m_i}}(\tau_i(0), x^0_{\ell_{k_i}^{m_i}}), \phi, c_i) < 0$, if $c_i = -1$.

Problem 1: Given a labeled set of time-robust tube segments $\tilde{S} = \{(R_{tube}(k_i, m_i, [t+\bar{a}_i, t+\bar{b}_i]), c_i)\}_{i=1}^N$ (the external time t flows from 0 to T) from a hybrid system $\mathcal{H}$, find an MTL formula ϕ such that ϕ evaluated at external time 0 perfectly classifies the trajectory segments with label $c_i = 1$ and the trajectory segments with label $c_i = -1$ in $\tilde{S}$, i.e. if $(\tau_i(t), \xi_{\ell_{k_i}^{m_i}}(\tau_i(t), \tilde{x}^0_{\ell_{k_i}^{m_i}})) \in R_{tube}(k_i, m_i, [t+\bar{a}_i, t+\bar{b}_i])$ for any $t \in [0,T]$, then $\hat{r}(\xi_{\ell_{k_i}^{m_i}}(\tau_i(0), \tilde{x}^0_{\ell_{k_i}^{m_i}}), \phi, c_i) > 0$, if $c_i = 1$; $\hat{r}(\xi_{\ell_{k_i}^{m_i}}(\tau_i(0), \tilde{x}^0_{\ell_{k_i}^{m_i}}), \phi, c_i) < 0$, if $c_i = -1$.

If for each location ℓ, the continuous dynamics is affine and stable, then there exists a quadratic autobisimulation function $\Phi_\ell(\xi_\ell(\tau, x_\ell^0), \xi_\ell(\tau, x)) = [(\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, x))^T M_\ell(\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, x))]^{\frac{1}{2}}$, where M_ℓ is positive definite. To solve problem 1, we first give the following three propositions:

Proposition 2: For any MTL formula ϕ and $\gamma_\ell > 0$, if $\Phi_\ell(\xi_\ell(\tau, \tilde{x}_\ell^0), \xi_\ell(\tau, x_\ell^0)) = [(\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, \tilde{x}_\ell^0))^T M_\ell (\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, \tilde{x}_\ell^0))]^{\frac{1}{2}} < \gamma_\ell$ for any $\tau \geq 0$, then for any τ (here τ can be positive or negative), $\hat{r}(\xi_\ell(\tau, x_\ell^0), \phi, c) - \hat{\gamma}_\ell \leq \hat{r}(\xi_\ell(\tau, \tilde{x}_\ell^0), \phi, c) \leq \hat{r}(\xi_\ell(\tau, x_\ell^0), \phi, c) + \hat{\gamma}_\ell$ holds, where c is a classification label, $\hat{\gamma}_\ell = \gamma_\ell \left\|M_\ell\right\|^{-\frac{1}{2}}$.

Proposition 3: For any MTL formula ϕ that only contains one variable x_j ($j = 1, 2, \ldots, n$) and $\gamma_\ell > 0$, if $\Phi_\ell(\xi_\ell(\tau, \tilde{x}_\ell^0), \xi_\ell(\tau, x_\ell^0)) = [(\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, \tilde{x}_\ell^0))^T M_\ell (\xi_\ell(\tau, x_\ell^0) - \xi_\ell(\tau, \tilde{x}_\ell^0))]^{\frac{1}{2}} < \gamma_\ell$ for any $\tau \geq 0$, and if there exists $z_{\ell,j} > 0$ such that $z_{\ell,j}^2 e_j e_j^T \preceq M_\ell$ (e_j is a canonical unit vector), then for any τ (here τ can be positive or negative), $\hat{r}(\xi_\ell(\tau, x_\ell^0), \phi, c) - \tilde{\gamma}_{\ell,j} \leq \hat{r}(\xi_\ell(\tau, \tilde{x}_\ell^0), \phi, c) \leq \hat{r}(\xi_\ell(\tau, x_\ell^0), \phi, c) + \tilde{\gamma}_{\ell,j}$ holds, where c is the classification label, $\tilde{\gamma}_{\ell,j} = \gamma_\ell / z_{\ell,j}$.

Proposition 4: Given the settings of Problem 1, an MTL formula ϕ evaluated at external time 0 perfectly classifies the trajectory segments with label $c_i = 1$ and the trajectory segments with label $c_i = -1$ in $\tilde{S}$ if the following condition is satisfied:
$MG(k_i, m_i, \bar{a}_i, \bar{b}_i, \phi, c_i) > 0$, if $c_i = 1$; $MG(k_i, m_i, \bar{a}_i, \bar{b}_i, \phi, c_i) < 0$, if $c_i = -1$, where $MG(\cdot)$ is a margin function defined as follows:

$$\begin{aligned} MG(k, m, \bar{a}, \bar{b}, \phi, 1) &= \min_{\tau \in [\bar{a}, \bar{b}]} \hat{r}(\xi_{\ell_k^m}(\tau, x_{\ell_k^m}^0), \phi, 1) - \hat{\gamma}_{\ell_k^m}, \\ MG(k, m, \bar{a}, \bar{b}, \phi, -1) &= \max_{\tau \in [\bar{a}, \bar{b}]} \hat{r}(\xi_{\ell_k^m}(\tau, x_{\ell_k^m}^0), \phi, -1) + \hat{\gamma}_{\ell_k^m}, \end{aligned} \tag{5}$$

where $\hat{\gamma}_{\ell_k^m} = \gamma_{\ell_k^m} \left\|M_{\ell_k^m}\right\|^{-\frac{1}{2}}$, $\gamma_{\ell_k^m}$ is obtained from (2) in each location ℓ_k^m.

According to Proposition 4, we can solve Problem 1 by minimizing the following cost function:

$$J(\tilde{S}, \phi) = \sum_{i=1}^{N} G(k_i, m_i, \bar{a}_i, \bar{b}_i, \phi, c_i), \tag{6}$$

where $G(\cdot)$ is defined as follows:

$$G(k, m, \bar{a}, \bar{b}, \phi, c) = \begin{cases} 0, & \text{if } c \cdot MG(k, m, \bar{a}, \bar{b}, \phi, c) > 0, \\ \zeta, & \text{otherwise}, \end{cases}$$

where the margin function $MG(\cdot)$ is defined in (5), ζ is a positive constant. When the MTL formula ϕ only contains one variable x_j, $\hat{\gamma}_\ell$ can be replaced by $\tilde{\gamma}_{\ell,j}$ in (5).

The core of the classification process is a non-convex optimization problem for finding the structure and the parameters that describe the MTL formula ϕ, which can be solved through particle swarm optimization [17]. The search starts from a basis of candidate formulae in the form of $\Box_{[\tau_1,\tau_2]}\pi$ or $\Diamond_{[\tau_1,\tau_2]}\pi$ and adding Boolean connectives until a satisfactory formula is found.

Once the optimization procedure obtains an optimal formula ϕ^* that perfectly classifies the labeled set of time-robust tube segments $\tilde{S}$ in each time period before the basic observer updates its states, we can use the obtained ϕ^* to refine the basic observer. The refinement procedure is to shrink the observer's state (i.e., the state estimate for $\mathcal{H}$) and the subsequent transitions based on satisfaction or violation of a MTL formula. For a given ϕ, we can shrink the observer state as soon as ϕ is satisfied or violated, while not resetting the timer. Then the subsequent states and transitions can be modeled in the same way as the basic observer as constructed in [12].

IV. Implementation

In this section, we implement our occupancy detection method to distinguish between two cases in the simulation model of a smart building testbed [18]: (i) one person enters an empty room after the door opens; (ii) two people enter an empty room after the door opens. We assume that we can observe the event when the door opens. The air conditioning is programmed to increase the mass flow rate of the cooling air when the temperature reaches certain thresholds (e.g. 290.6K, 290.7K). The system is modeled as a hybrid system $\mathcal{H}$ with 6 locations, as shown in Fig. 1. The state $x = [T, w, \dot{Q}_{\rm gen}, \dot{W}_{\rm gen}]$ represents the temperature and humidity ratio of the room, heat and humidity generation rate within the room (i.e. from the humans) respectively (we choose the units of $\dot{Q}_{\rm gen}$ and $\dot{W}_{\rm gen}$ to be W and mg/s, respectively). $\dot{Q}_{\rm gen}$ and $\dot{W}_{\rm gen}$ are added as two pseudo-states to account for the variations of the heat and humidity generation rates by different people [19]. The continuous dynamics in the 6 locations are given as follows:

For location ℓ^0 (room unoccupied):

$$\begin{cases} C\dot{x}_1 = \dot{m}_{\ell^0} C_{\rm p}(T_{\rm s} - x_1) + \beta G(x_2 - w_\infty) - K(x_1 - T_\infty); \\ M\dot{x}_2 = \dot{m}(w_{\rm s} - x_2) - G(x_2 - w_\infty); \\ \dot{x}_3 = 0; \dot{x}_4 = 0. \end{cases}$$

For the other 5 locations ℓ_k^m (ℓ_1^1, ℓ_1^2, ℓ_2^1, ℓ_2^2, ℓ_2^3, room occupied with one or two people):

$$\begin{cases} C\dot{x}_1 = \dot{m}_{\ell_k^m} C_{\rm p}(T_{\rm s} - x_1) + \beta G(x_2 - w_\infty) - K(x_1 - T_\infty) \\ \qquad + x_3 - 10^{-6}\beta x_4; \\ M\dot{x}_2 = \dot{m}_{\ell_k^m}(w_{\rm s} - x_2) - G(x_2 - w_\infty) + x_4; \\ \dot{x}_3 = 0; \dot{x}_4 = 0. \end{cases}$$

where $\dot{m}_{\ell_k^m}$ is the mass flow rate of the air conditioning in location ℓ_k^m (we set $\dot{m}_{\ell^0} = \dot{m}_{\ell_1^1} = \dot{m}_{\ell_2^1} = 0.5$Kg/s, $\dot{m}_{\ell_1^2} = \dot{m}_{\ell_2^2} = 0.6$Kg/s, $\dot{m}_{\ell_2^3} = 0.8$Kg/s), C is the thermal capacitance of the room, M is mass of air in the room, G is the mass transfer conductance between the room and the ambient, $w_{\rm s}$, $T_{\rm s}$ are the supply air humidity ratio and temperature respectively, w_∞, T_∞ are the ambient humidity ratio and temperature respectively, $C_{\rm p}$ is specific heat of air at constant pressure, β is latent heat of vaporization of water, K is the wall thermal conductance.

We set $T_\infty = 303$K (29.85°C), $T_{\rm s} = 290$K (16.85°C), $w_\infty = 0.0105$, $w_{\rm s} = 0.01$. As shown in Fig. 2, as human can generate both heat and moisture, the room temperature and humidity ratio will increase towards the new equilibrium after people enter the room. As the mass flow rate of the air conditioning may change in different locations, when two people enter the empty room, the temperature first increases to 290.7K, then starts to decrease as the mass flow rate is increased to 0.8Kg/s. It can be seen that the steady state values of the temperatures in the two cases are almost the same, therefore a temporal logic formula is needed to distinguish their temporal patterns in the transient period.

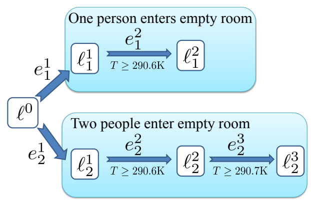

Fig. 1. Locations of hybrid system $\mathcal{H}$ for the smart building model describing the series of events of the two cases.

The invariant sets are

$$\begin{aligned} Inv(\ell^0) &= \mathbb{R}^4, \\ Inv(\ell_1^1) &= Inv(\ell_2^1) = \{x | 290.4 \leq x_1 \leq 290.6\}, \\ Inv(\ell_1^2) &= Inv(\ell_2^2) = \{x | 290.5 \leq x_1 \leq 290.7\}, \\ & \quad\; Inv(\ell_2^3) = \{x | 290.6 \leq x_1 \leq 290.8\}. \end{aligned}$$

The events are modeled as follows:

- $e_1^1 = (\ell^0, \ell_1^1, g_1^1, r_1^1)$, where $g_1^1 = \mathbb{R}^4$, $r_1^1(x) = x + [0, 0, 300, 80]$;
- $e_2^1 = (\ell^0, \ell_2^1, g_2^1, r_2^1)$, where $g_2^1 = \mathbb{R}^4$, $r_2^1(x) = x + [0, 0, 600, 160]$;
- $e_1^2 = (\ell_1^1, \ell_1^2, g_1^2, r_1^2) = e_2^2 = (\ell_2^1, \ell_2^2, g_2^2, r_2^2)$, where $g_1^2 = g_2^2 = \{x | x_1 = 290.6\}$, $r_1^2(x) = r_2^2(x) = x$;
- $e_2^3 = (\ell_2^2, \ell_2^3, g_2^3, r_2^3)$, where $g_2^3 = \{x | x_1 = 290.7\}$, $r_2^3(x) = x$.

The events e_1^1 and e_2^1 are non-deterministic, i.e. the events can happen anywhere in $Inv(\ell^0)$; the events e_1^2, e_2^2 and e_2^3 are deterministic, i.e. the events are forced to occur whenever the states leave the invariant sets (reach the guards). The output symbols of events e_1^1 and e_2^1 are observable (door

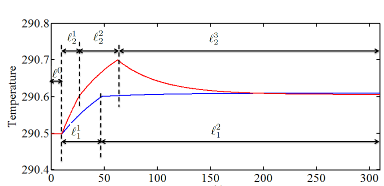

Fig. 2. The temperature state of the two simulated trajectories (blue represents the trajectory when one person enters the empty room, red represents the trajectory when two people enter the empty room) and the corresponding locations.

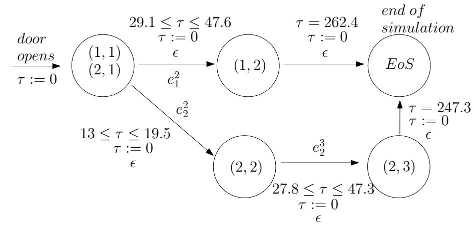

Fig. 3. A timed abstraction of the hybrid automaton $\mathcal{H}$. τ is the clock time that is associated with each trajectory which is reset to zero every time the trajectory enters a new location. For instance, the transition from $(1,1)$ to $(1,2)$ means that any trajectory of $\mathcal{H}$ initiated from $B_{\ell_1^1}(\gamma_{\ell_1^1}, x^0_{\ell_1^1})$ will reach $B_{\ell_1^2}(\gamma_{\ell_1^2}, x^0_{\ell_1^2})$ within 29.1 to 47.6 time units by triggering an unobservable event.

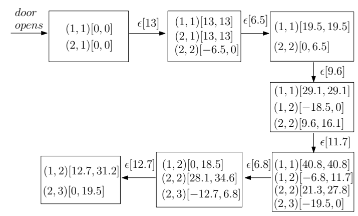

Fig. 4. The basic observer is driven by observed events and an external timer. Each block represents the observer state and it is updated to the next block at the time when an unobservable event becomes possible to be triggered, or an unobservable event is deduced to have been triggered, or an observable event is not possible to occur anymore [12]. When the observer state is s, the external time is t, the clock time and the state of $\mathcal{H}$ should be in $\{(\tau_k^m, \ell_k^m, x) | (\tau_k^m, x) \in R_{tube}(k, m, [t+\bar{a}, t+\bar{b}]), (k,m)[\bar{a}, \bar{b}] \in s\}$.

opening) while the output symbols of events e_1^2, e_2^2 and e_2^3 are unobservable.

The reset initial state at location ℓ_1^1 lies in the following set:

$$\begin{aligned} \mathcal{L}_1^1 \times \mathcal{X}_1^1 = \{\ell_1^1\} \times \{x \quad | & \quad x_1 = 290.4976, x_2 = 0.01, \\ & 280 \leq x_3 \leq 320, 60 \leq x_4 \leq 100\}. \end{aligned}$$

The reset initial state at location ℓ_2^1 lies in the following set:

$$\begin{aligned} \mathcal{L}_2^1 \times \mathcal{X}_2^1 = \{\ell_1^1\} \times \{x \quad | & \quad x_1 = 290.4976, x_2 = 0.01, \\ & 560 \leq x_3 \leq 640, 120 \leq x_4 \leq 200\}. \end{aligned}$$

We use the MATLAB Toolbox STRONG [20] to simulate two trajectories for the two cases. As the variation range of the temperature is much smaller than the variation ranges of the humidity and heat generation rates, in order to cover the reset initial sets $\mathcal{L}_1^1 \times \mathcal{X}_1^1$ and $\mathcal{L}_2^1 \times \mathcal{X}_2^1$, we optimize the matrix M_ℓ in each location ℓ (geometrically change the shape of the level set ellipsoid) so that the outer bounds of the level set ellipsoid $B_\ell(\gamma_\ell, x_\ell^0)$ in the dimension of the temperature variation is much smaller than the outer bounds in the other dimensions. Besides, according to Proposition 3, we use the tighter bound $\tilde{\gamma}_{\ell,1} = \gamma_\ell / z_{\ell,1}$ for the optimization for MTL classification (we use the data of the simulated room temperature to infer the MTL formula and the case for the room humidity ratio can be done in a similar manner), and by maximizing $z_{\ell,1}$ thus minimizing $\tilde{\gamma}_{\ell,1}$, we can obtain the tightest bound $\tilde{\gamma}^*_{\ell,1} = \gamma_\ell / z^*_{\ell,1}$. The combined optimization to obtain both M^*_ℓ and $z^*_{\ell,1}$ is as follows:

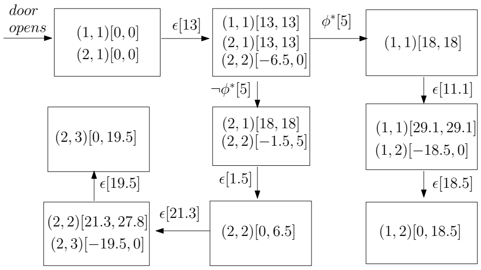

Fig. 5. The refined observer shrink the basic observer's states by adding the inferred MTL formula ϕ^* in (8) (ϕ^* is evaluated at external time 0, i.e. at the time instant the basic observer updates its states). The satisfaction and violation of ϕ are modeled as transition labels $\phi^*[5]$ and $\neg\phi^*[5]$ respectively, where 5 is the minimal time needed for evaluating the truth value of ϕ^*.

$$\begin{aligned} &\min. -z_{\ell,1}^2 \\ \text{s.t. } & M_\ell \succ 0, A_\ell^T M_\ell + M_\ell A_\ell \prec 0, \\ & e_3^T M_\ell e_3 \le \eta_3, e_4^T M_\ell e_4 \le \eta_4, \\ & e_1^T M_\ell e_1 \ge \eta_1, M_\ell - z_{\ell,1}^2 e_1 e_1^T \succeq 0. \end{aligned} \tag{7}$$

where A_ℓ is the state (or system) matrix in location ℓ, $e_1 = [1,0,0,0]^T, e_3 = [0,0,1,0]^T, e_4 = [0,0,0,1]^T$, $\eta_1 = 30$, $\eta_3 = \eta_4 = 10^{-7}$ (η_1, η_3 and η_4 are tuned manually for covering the reset initial sets $\mathcal{L}_1^1 \times \mathcal{X}_1^1$ and $\mathcal{L}_2^1 \times \mathcal{X}_2^1$).

The optimal solution is computed as $z_{\ell,1}^* = 30$. Based on the two simulated trajectories, we construct a timed abstraction (timed automaton) as shown in Fig. 3 (for details of constructing the timed automaton, see the content of timed abstraction in [12]). All the events are unobservable except ψ which represents the door opening. We construct a basic observer as in Fig. 4 (for details of designing the basic observer, see [12]), where the two occupancy states are never distinguished to the end of the simulation time.

Next we infer an MTL formula that classifies the time-robust tube segments corresponding to the basic observer's states. The observer's initial state s^1 contains $(1,1)[0,0]$ and $(2,1)[0,0]$. We first classify the time-robust tube segment $R_{tube}(1,1,[t,t])$ and $R_{tube}(2,1,[t,t])$ (t flows from 0 to 13) but does not find any MTL formula that can achieve perfect classification. Then we move on to state s^2 which contains $(1,1)[13,13]$, $(2,1)[13,13]$ and $(2,2)[-6.5,0]$. We find the following formula that perfectly clasifies $R_{tube}(1,1,[t,t])$ from $R_{tube}(2,1,[t,t])$ and $R_{tube}(2,2,[t-6.5,t])$ (t flows from 0 to 6.5):

$$\phi^* = \Box_{[1.7717,5]}(x_1 \ge 290.6006). \tag{8}$$

The optimization takes 36.7 seconds on a Thinkpad laptop computer with Intel Core i7 and 8GB RAM.

With the inferred MTL formula ϕ^*, we construct the refined observer as shown in Fig. 5. It can be seen that once ϕ^* is satisfied, the two cases are distinguished in 18 seconds. Compared with the basic observer which can never distinguish the two occupancy states, the refinement has achieved the result in 18 seconds by only adding one temperature sensor.

References

[1] E. Hailemariam, R. Goldstein, R. Attar, and A. Khan, "Real-time occupancy detection using decision trees with multiple sensor types," in *Proc. Symposium on Simulation for Architecture and Urban Design*, ser. SimAUD '11. San Diego, CA, USA: Society for Computer Simulation Int., 2011, pp. 141–148. [Online]. Available: http://dl.acm.org/citation.cfm?id=2048536.2048555

[2] Q. Hua, H. B. Chen, Y. Y. Ye, and S. X. D. Tan, "Occupancy detection in smart buildings using support vector regression method," in *Proc. Int. Conf. on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 02, Aug 2016, pp. 77–80.

[3] R. Tomastik, S. Narayanan, A. Banaszuk, and S. Meyn, *Model-Based Real-Time Estimation of Building Occupancy During Emergency Egress*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 215–224.

[4] E. Asarin, A. Donzé, O. Maler, and D. Nickovic, "Parametric identification of temporal properties," in *Proc. Int. Conf. Runtime Verification*, Berlin, Heidelberg, 2012, pp. 147–160.

[5] B. Hoxha, A. Dokhanchi, and G. Fainekos, "Mining parametric temporal logic properties in model-based design for cyber-physical systems," *Int. Journal on Software Tools for Technology Transfer*, Feb 2017. [Online]. Available: http://dx.doi.org/10.1007/s10009-017-0447-4

[6] X. Jin, A. Donze, J. V. Deshmukh, and S. A. Seshia, "Mining requirements from closed-loop control models," in *Proc. Int. Conf. Hybrid Systems: Computation and Control*, 2013, pp. 43–52.

[7] Z. Kong, A. Jones, and C. Belta, "Temporal logics for learning and detection of anomalous behavior," *IEEE Trans. Autom. Control*, vol. 62, no. 3, pp. 1210–1222, March 2017.

[8] Z. Xu, M. Birtwistle, C. Belta, and A. Julius, "A temporal logic inference approach for model discrimination," *IEEE Life Sciences Letters*, vol. 2, no. 3, pp. 19–22, Sept 2016.

[9] Z. Xu and A. A. Julius, "Census signal temporal logic inference for multiagent group behavior analysis," *IEEE Trans. Autom. Sci. and Eng.*, Early Access on IEEE Xplore.

[10] G. Bombara, C.-I. Vasile, F. Penedo, H. Yasuoka, and C. Belta, "A decision tree approach to data classification using signal temporal logic," in *Proc. Int. Conference on Hybrid Systems: Computation and Control*, ser. HSCC '16. New York, NY, USA: ACM, 2016, pp. 1–10. [Online]. Available: http://doi.acm.org/10.1145/2883817.2883843

[11] Z. Xu, S. Saha, and A. Julius, "Provably correct design of observations for fault detection with privacy preservation," in *Proc. IEEE Conf. Decision and Control (CDC), Melbourne, Australia, 2017*.

[12] Y. Deng, A. D'Innocenzo, and A. A. Julius, "Trajectory-based observer for hybrid automata fault diagnosis," in *Proc. IEEE Conf. Decision and Control (CDC)*, Dec 2015, pp. 942–947.

[13] R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P. H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine, "The algorithmic analysis of hybrid systems," *Theoretical Computer Science*, vol. 138, pp. 3–34, 1995.

[14] A. A. Julius, G. E. Fainekos, M. Anand, I. Lee, and G. J. Pappas, "Robust test generation and coverage for hybrid systems," in *Proc. Hybrid Syst.: Computat. Control*. Springer, 2007, pp. 329–342.

[15] A. Girard, "Approximately bisimilar finite abstractions of stable linear systems," in *Proc. Hybrid Syst.: Comput. and Control*, Pisa, Italy, 2007.

[16] Z. Xu, A. Julius, and J. H. Chow, "Energy storage controller synthesis for power systems with temporal logic specifications," *IEEE Systems Journal*, Early access on IEEE Xplore.

[17] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, vol. 4, Nov 1995, pp. 1942–1948 vol.4.

[18] C. C. Okaeme, S. Mishra, and J. T. Wen, "A comfort zone set-based approach for coupled temperature and humidity control in buildings," in *Proc. IEEE Int. Conf. Automation Science and Engineering (CASE)*, Aug 2016, pp. 456–461.

[19] A. TenWolde and C. L. Pilon, "The effect of indoor humidity on water vapor release in homes," in *Proc. Int. Conf. Thermal Performance of the Exterior Envelopes of Whole Buildings X*, Dec 2007.

[20] Y. Deng, A. Rajhans, and A. A. Julius, "Strong: A trajectory-based verification toolbox for hybrid systems," in *Quantitative Evaluation of Systems*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2013, vol. 8054, pp. 165–168.